UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/713,602	11/14/2003	Bryan M. Cantrill	03226.349001; SUN040251	6953

32615 7590 02/22/2006

OSHA LIANG L.L.P./SUN
1221 MCKINNEY, SUITE 2800
HOUSTON, TX 77010

EXAMINER	
CAMPOS, YAIMA	
ART UNIT	**PAPER NUMBER**
2185	

DATE MAILED: 02/22/2006

Please find below and/or attached an Office communication concerning this application or proceeding.

Office Action Summary	Application No.	Applicant(s)	
	10/713,602	CANTRILL, BRYAN M.	
	Examiner	Art Unit	
	Yaima Campos	2185	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on *14 November 2003*.

2a)☐ This action is **FINAL**. 2b)☒ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *1-20* is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *1-20* is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☐ The specification is objected to by the Examiner.

10)☒ The drawing(s) filed on *14 November 2003* is/are: a)☒ accepted or b)☐ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☐ All b)☐ Some * c)☐ None of:

1.☐ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____.

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☒ Notice of References Cited (PTO-892)
2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
3) ☒ Information Disclosure Statement(s) (PTO-1449 or PTO/SB/08) Paper No(s)/Mail Date *2/17/04*.
4) ☐ Interview Summary (PTO-413) Paper No(s)/Mail Date. _____.
5) ☐ Notice of Informal Patent Application (PTO-152)
6) ☐ Other: _____.

DETAILED ACTION

1. The instant application having Application No. 10/713,602 has a total of 20 claims pending in the application; there are 2 independent claims and 18 dependent claims, all of which are ready for examination by the examiner.

I. INFORMATION CONCERNING OATH/DECLARATION

Oath/Declaration

2. The applicant's oath/declaration has been reviewed by the examiner and is found to conform to the requirements prescribed in **37 C.F.R. 1.63.**

II. INFORMATION CONCERNING DRAWINGS

Drawings

3. The applicant's drawings submitted are acceptable for examination purposes.

III. ACKNOWLEDGEMENT OF REFERENCES CITED BY APPLICANT

4. As required by **M.P.E.P. 609(C)**, the applicant's submission of the Information Disclosure Statement dated February 17, 2004 is acknowledged by the examiner and the cited references have been considered in the examination of the claims now pending. As required by **M.P.E.P 609 C(2)**, a copy of the PTOL-1449 initialed and dated by the examiner is attached to the instant office action.

IV. OBJECTIONS TO THE SPECIFICATION

Claim Objections

5. Claim 6 is objected to as being a substantial duplicate of claim 5.

6. Applicant is advised that should claim 5 be found allowable, claim 6 will be objected to under 37 CFR 1.75 as being a substantial duplicate thereof. When two claims in an application are duplicates or else are so close in content that they both cover the same thing, despite a slight difference in wording, it is proper after allowing one claim to object to the other as being a substantial duplicate of the allowed claim. See MPEP § 706.03(k).

7. Appropriate correction is required.

V. REJECTIONS BASED ON PRIOR ART

Claim Rejections - 35 USC § 103

8. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

9. **Claims 1-15 and 17-20** are rejected under 35 U.S.C. 103(a) as being unpatentable over Ayers et al. (US 6,804,814) in view of Huras et al. (US 2004/0181635) and further in view of Weng et al. (US 5,659,698).

10. As per **claims 1, 8 and 17**, Ayers discloses "A method/system for storing a data set having an enabled probe identification component and an associated data component in a buffer," as a system for tracing program execution that **["records data values**

loaded or stored by the program as well as the instructions in one or more circular buffers" (Column 1, lines 51-53) in which "a final value set can comprise system level parameters and values, such as but not limited to the names and identifiers of other processes running on the same machine at the time of recording" (Column 2, lines 62-65) and explains that "a probe is inserted into the program to save a value of a particular variable at a particular instruction in the program" (Column 3, lines 16-18)] comprising "a probe configured to obtain data from an instrumented program; a tracing framework configured to associate the probe with an enabled probe identification;" **[With respect to this limitation, Ayers discloses "the program is instrumented to record the value sets" (Column 2, lines 32-33) and explains that "special instrumentation probes can be used to specifically monitor the changes to such variable values that result from particular complex instructions or from invocations of code sequences that do not contain instrumentation that will reveal the exact sequence of statement executions" (Column 9, lines 58-63)]** "wherein the stored data set size is determined using an enabled probe identification associated with the stored data set" **[Ayers discloses this concept as "the instrumentation code added by the instrumentor created an instruction trace of the execution, recorded value sets, and a probe log containing information recorded by instrumented probes" (Column 10, lines 37-38) as the size of the recorded data may vary and "the quantity of data to be recorded is adjusted with a control such as a virtual dial shown on a display" (Column 3, lines 36-38)]**.

Ayers does not disclose expressly that "the data is stored in the data component and the enabled probe identification is stored in the enabled probe identification

component," nor "storing the data set at a current offset if the buffer has sufficient space to store the data set between a current offset and a limit of the buffer and the buffer is not marked as wrapped; marking the buffer as wrapped, setting the current offset to zero and setting a wrapped offset to zero, if the buffer does not have sufficient space to store the data set between a current offset and a limit of the buffer; and incrementing the wrapped offset by a stored data set size until there is sufficient space between the current offset and the wrapped offset to store the data set if the buffer is marked as wrapped."

Huras discloses that "the data is stored in the data component and the enabled probe identification is stored in the enabled probe identification component," as **[a buffer containing "region 402 comprises a control region 404, a data storing region 406 and a wasteland region 408" (Figure 4A and Column 4, paragraph 0041, lines 6-9)];** "storing the data set at a current offset if the buffer has sufficient space to store the data set between a current offset and a limit of the buffer and the buffer is not marked as wrapped;" **[With respect to this limitation, Huras discloses a "reservation system 306 is enabled to operate in wrap, non-wrap and dump modes" (Column 4, paragraph 0049). Huras further explains that "if buffer wrapping is disabled and if the buffer is full, then the index of the next available slot of the next reservation is equal to or grater than the maximum number of slots in the buffer, and buffer truncation occurs" (Column 6, paragraph 0066, lines 5-9) as demonstrating that when using a non-wrapping buffer mode, entries are added to a buffer until a maximum or limit of the buffer is reached]** "marking the buffer as wrapped," **[Huras discloses this concept as "buffer wrapping occurs in an incremental implementation when the value of the atomic sequence exceeds the total number (maxSlot) of slots in the**

buffer" (Column 5, paragraph 0050, lines 3-5); further explains that "the buffer may be wrapped and previous reservations may be overwritten by new reservations" (Column 7, paragraph 0075, lines 1-3) and also discloses a reset method wherein "the system atomically adjusts the sequence value to the value that defines the next reservation at zero slot index" (Column 7, paragraph 0072, lines 3-5). Huras also teaches this limitation as "the wrap region is used for storing a variable that indicates a status of buffer wrapping (e.g. active or not)" and specifies that "a non-zero value stored in the wrap region indicates that the buffer wrapping is enabled" (Column 5. paragraph 0053, lines 7-12)] "and incrementing the wrapped offset by a stored data set size until there is sufficient space between the current offset and the wrapped offset to store the data set if the buffer is marked as wrapped," **[With respect to this limitation, Huras discloses an example in which a buffer has slots 0-7 and "slot 7 remains available for reservation and the value of the next sequence variable is equal to seven. If a writer requests three slots, value of the next sequence variable may be atomically incremented to ten and the value of the slot index variable becomes equal to seven. A region comprising slots 7, 0 and 1 has been reserved for a writer" (Column 6, paragraph 0065, lines 9-18)]** but does not disclose expressly "setting the current offset to zero and setting a wrapped offset to zero, if the buffer does not have sufficient space to store the data set between a current offset and a limit of the buffer."

Weng discloses "setting the current offset to zero and setting a wrapped offset to zero, if the buffer does not have sufficient space to store the data set between a current offset and a limit of the buffer" as **["To generate a new address 650, the current**

address 644 and the offset value 646 are supplied to the adder 638" then "the offset value is added to the current value 644 to produce a potential new address 635. The potential new address is compared with the ending address 640 by the comparator 634. The comparator 634, compares, for an incrementing circular buffer, the ending address 640 with the potential new address 635. When the potential new address 635 is unfavorable, i.e., exceeds the ending address 640, the comparator 634 generates a second control signal 648. The second control signal 648 is supplied to the multiplexer 636 which instructs the multiplexer 636 to select the beginning address 642 as the new address 650" (Column 25, lines 1-18)].

Ayers et al. (US 6,804,814), Huras et al. (US 2004/0181635) and Weng et al. (US 5,659,698) are analogous art because they are form the same field of endeavor of data storage in buffer memory.

At the time of the invention it would have been obvious to a person of ordinary skill in the art to combine the method/system for tracing program execution that stores traced data in a circular buffer as taught by Ayers further manage the circular buffer to have a wrapping mode as well as a non-wrapping mode as taught by Huras and further "setting the current offset to zero and setting a wrapped offset to zero, if the buffer does not have sufficient space to store the data set between a current offset and a limit of the buffer" when using buffer wrap mode as taught by Weng.

The motivation for doing so would have been because Huras teaches that managing a circular buffer to have a wrapping mode as well as a non-wrapping mode **["facilitates synchronization and thus reduces serialization" (Column 6, paragraph 0065, lines 22-24). Huras also explains that "a reservation system in accordance with**

the present invention serves to reduce writer synchronization in an application sharing memory among a plurality of writers. The use of an atomic counter to indicate the next available portion of memory" ensures that "individual writers are temporarily assigned unique portions of the memory efficiently" (Column 7, paragraph 0077)] and because Weng teaches that **["when the potential next address compared favorably with the ending address, use the potential next address as a new address, otherwise generate the new address from the beginning address of the circular buffer. With such a method and apparatus, a 2.sup.K limitation of prior art circular buffer address generation unit is eliminated, thus reducing memory requirements" (Column 4, lines 10-24)]**.

Therefore, it would have been obvious to combine Weng et al. (US 5,659,698) with Ayers et al. (US 6,804,814) and Huras et al. (US 2004/0181635) for the benefit of creating a method/system for storing program execution tracing data to obtain the invention as specified in claims 1, 8 and 17.

11. As per **claims 2 and 9**, the combination of Ayers, Huras and Weng discloses "The method of claim 1," **[See rejection to claim 1 above]** further comprising "storing the data set at the current offset if there is sufficient space between the current offset and the wrapped offset and the buffer is marked as wrapped" as **[Weng discloses this limitation as "to generate a new address, 650 the comparator 634 compares the limiting address 660 with the current address 644" (Column 25, lines 45-63) and "when the comparison between the current address 644 and the limiting address 660 is favorable, i.e. the current address 644 is less the limiting address 660, the comparator generates the first control signal 648. Upon receiving the first control**

signal, the multiplexer 636 selects, as the new address 650, the resultant of the adder 638. The resultant of the adder 638 is the summation of the current address 634 with the offset value 646" (Columns 25-26, lines 64-67 and 1-6)].

12. As per **claim 3**, the combination of Ayers, Huras and Weng disclose "The method of claim 2," **[See rejection to claim 2 above]** "further comprising: incrementing the current offset by a data set size after the data set has been stored" **[With respect to this limitation Huras discloses that "the shared memory buffer is divided into numerous fixed size regions called *slots*. Reservations are always made in multiples of slots. An atomic sequence counter is used to indicate the next available slot. A writer makes a reservation by incrementing the atomic sequence counter by the number of slots requested" (Column 3, paragraph 0038)].**

13. As per **claim 4**, the combination of Ayers, Huras and Weng discloses "The method of claim 3," **[See rejection to claim 3 above]** "further comprising: invalidating a buffer space between the current offset and the wrapped offset" **[With respect to this limitation, Huras discloses that when "buffer wrapping is enabled, a sequence value is atomically incremented by maxSlots to invalidate all previous reservations in the buffer. After the sequence value is incremented, only new reservations may be considered valid and all other reservations may be considered invalid" (Column 7, paragraph 0071)].**

14. As per **claims 5 and 6** , the combination of Ayers, Huras and Weng discloses "The method of claim 4," **[See rejection to claim 4 above]** "wherein the buffer space is invalidated by assigning each word within the buffer space a reserved enabled probe identification denoting a zero-length data component" **[Huras discloses an equivalent**

system/method with functionality to "calculate the memory requirements for a buffer; reserve a region of memory within a buffer; synchronize changes to the reserved region of memory; invalidate all the current contents of the buffer; dump the contents of the entire buffer" (Column 3, paragraph 0039). Huras also discloses that "the reservation system is enabled to clear or reset the buffer in order to remove the existing reservations" (Column 4, paragraph 0048, lines 1-3)].

15. As per **claim 7**, the combination of Ayers and Huras discloses "The method of claim 1," **[See rejection to claim 1 above]** "further comprising: invalidating a buffer space between the current offset and the limit of the buffer if the buffer is not wrapped and the buffer does not have sufficient space to store the data set between a current offset and a limit of the buffer" **[Huras discloses this concept as "if buffer wrapping is disabled and if the buffer is full, then the index of the next available slot of the next reservation is equal to or greater than the maximum number of slots (maxSlot) in the buffer, and buffer truncation occurs. Once the buffer is full the reservation system prevents further reservations by flagging the truncated region" (Column 6, paragraph 0066, lines 5-11)].**

16. As per **claim 10**, the combination of Ayers, Huras and Weng discloses "The method of claim 1," **[See rejection to claim 1 above]** but fail to disclose expressly "determining whether the wrapped offset is equal to the limit of the buffer;" **[Weng discloses this limitation as "to generate a new address, 650 the comparator 634 compares the limiting address 660 with the current address 644" (Column 25, lines 45-63)]** "storing the data set at the current offset if there is sufficient space between the current offset and the wrapped offset, the buffer is marked as wrapped, and the wrapped

offset is not equal to the limit of the buffer;" **["when the comparison between the current address 644 and the limiting address 660 is favorable, i.e. the current address 644 is less the limiting address 660, the comparator generates the first control signal 648. Upon receiving the first control signal, the multiplexer 636 selects, as the new address 650, the resultant of the adder 638. The resultant of the adder 638 is the summation of the current address 634 with the offset value 646" (Columns 25-26, lines 64-67 and 1-6)** "and setting the wrapped offset to zero and storing the data set at the current offset if the buffer is marked as wrapped and the wrapped offset is equal to the limit of the buffer" **[With respect to this limitation, Weng discloses; "To generate a new address 650, the current address 644 and the offset value 646 are supplied to the adder 638" then "the offset value is added to the current value 644 to produce a potential new address 635. The potential new address is compared with the ending address 640 by the comparator 634. The comparator 634, compares, for an incrementing circular buffer, the ending address 640 with the potential new address 635. When the potential new address 635 is unfavorable, i.e., exceeds the ending address 640, the comparator 634 generates a second control signal 648. The second control signal 648 is supplied to the multiplexer 636 which instructs the multiplexer 636 to select the beginning address 642 as the new address 650" (Figure 6 and Column 25, lines 1-18)]**.

17. As per **claims 11 and 18,** the combination of Ayers, Huras and Weng discloses "The method of claim 1," **[See rejection to claim 1 above]** "further comprising defining a tracing function wherein the tracing function comprises an action; associating the action with the enable probe identification and associating the probe with the enabled probe

identification" **[Ayers discloses this concept as "a method for creating a program execution data trace, comprises recording a first value set associated with the execution of a fist instruction referenced in an instruction trace" (Column 2, lines 24-27) wherein "the program is instrumented to record the value sets" (Column 2, lines 32-33) and explains that "a probe is inserted into a program to save a value of a particular variable at a particular instruction in the program" (Column 3, lines 16-18). Ayers further discloses accessing "a symbol table or an extended range table" to "retrieve a variable's name" (Column 3, lines 45-46)]**.

18. As per **claim 12**, the combination of Ayers, Huras and Weng discloses "The method of claim 11," **[See rejection to claim 11 above]** "wherein the tracing function is defined by a consumer" **[With respect to this limitation, Ayers discloses that "the control can allow a user to, for example, set the time interval after which data is recorded, or alternatively, to se the frequency of a predetermined event at which to record data, or alternatively to set the type of data to be recorded, or to set address ranges within which to record data" (Column 3, lines 38-43)]**.

19. As per **claim 13 and 20**, the combination of Ayers, Huras and Weng discloses "The method of claim 11," **[See rejection to claim 11 above]** "wherein the enabled probe identification is defined on a per-consumer basis" **[With respect to this limitation Ayers discloses that "an input device permits a user to request a value of a data variable corresponding to a particular instruction in the instruction trace" (Column 4, lines 4-6) and also explains that "variable values displayed in the data log can also be placed next to their user visible name from the program to aid in understandability" (Columns 10-11, lines 66-67 and 1). Ayers also teaches; "a probe**

is interted into the program to save a value of a particular variable at a particular instruction in the program" (Column 3, lines 16-18)].

20. As per **claims 14-15 and 19,** the combination of Ayers and Huras discloses "The method of claim 1," **[See rejection to claim 1 above]** "wherein the enabled probe identification associated with the stored data set is used as a reference in to an enabled probe identification-metadata table" **[With respect to this limitation, Ayers discloses that "in one embodiment, a table is maintained which associates program instructions encountered in the instruction trace with simulation instructions which reverse the operation of the associated program instructions" (Column 2, lines 47-50); and also teaches that "a probe is interted into the program to save a value of a particular variable at a particular instruction in the program" (Column 3, lines 16-18) wherein "a symbol table or an extended range table is accessed to retrieve a variable's name. The variable's name is then displayed next to the variable's value" (Column 3, lines 45-48)].**

21. **Claim 16** is rejected under 35 U.S.C. 103(a) as being unpatentable over Ayers et al. (US 6,804,814), Huras et al. (US 2004/0181635) and Weng et al. (US 5,659,698) as applied to claims 1-15 and 17-20 above, and further in view of Borg et al. (US 5,274,811).

22. As per **claim 16**, the combination of Ayer, Huras and Weng discloses "The method of claim 1," **[See rejection to claim 1 above]** but fails to disclose expressly that "the data set is stored in a kernel-level buffer."

Borg discloses storing trace data in a kernel-level buffer as **["the trace routine record an instruction or data reference in a predefined long trace buffer, the trace**

buffer having been designated when the kernel was booted" (Column 3, lines 26-29)].

Ayers et al. (US 6,804,814), Huras et al. (US 2004/0181635), Weng et al. (US 5,659,698) and Borg et al. (US 5,274,811) are analogous art because they are form the same field of endeavor of data storage in buffer memory.

At the time of the invention it would have been obvious to a person of ordinary skill in the art to combine the method/system for tracing program execution that stores traced data in a circular buffer as taught by Ayers further manage the circular buffer to have a wrapping mode as well as a non-wrapping mode as taught by Huras, further "setting the current offset to zero and setting a wrapped offset to zero, if the buffer does not have sufficient space to store the data set between a current offset and a limit of the buffer" when using buffer wrap mode as taught by Weng and further specifically use a kernel buffer for storing trace data as taught by Borg.

The motivation for doing so would have been because Borg teaches; **["The operating system kernel was modified to allocate the trace buffer, and to cause it to be mapped into the address space of every user process linked by the linking loader. In this way, specially linked user processes can write trace information into the buffer without trapping into the kernel. Since the kernel and the users access the same buffer, the entries made by the kernel and by users are interleaved in precisely the order they are actually executed" (Column 3, lines 42-50)]**.

Therefore, it would have been obvious to combine Weng et al. (US 5,659,698) with Ayers et al. (US 6,804,814) and Huras et al. (US 2004/0181635) for the benefit of

creating a method/system for storing program execution tracing data to obtain the invention as specified in claim 16.

VI. RELEVANT ART CITED BY THE EXAMINER

23. The following prior art made of record and not relied upon is cited to establish the level of skill in the applicant's art and those arts considered reasonably pertinent to applicant's disclosure. See **MPEP 707.05(c)**.

24. The following references teach tracing program execution and storing tracing data.

U.S. PATENT NUMBER

US 6,282,701

US 5,737,521

US 6,321,290

US 6,173,395

US 6,247,146

25. The following references teach storing and accessing data in a circular buffer.

U.S. PATENT NUMBER

US 6,584,556

US 6,363,470

US 2002/0156990

US 4,800,524

US 2003/0033499

26. The following reference teaches a method of tracing program execution that supports three modes of collecting data in a buffer (fill, wrap or contiguous).

REFERENCE IDENTIFICATION

IBM Technical Disclosure Bulletin Cross Reference: 0374-4353-0-431-538

VII. CLOSING COMMENTS

Conclusion

a. STATUS OF CLAIMS IN THE APPLICATION

27. The following is a summary of the treatment and status of all claims in the application as recommended by **M.P.E.P. 707.07(i)**:

a(1) CLAIMS REJECTED IN THE APPLICATION

28. Per the instant office action, claims 1-20 have received a first action on the merits and are subject of a first action non-final.

b. DIRECTION OF FUTURE CORRESPONDENCES

29. Any inquiry concerning this communication or earlier communications from the examiner should be directed to Yaima Campos whose telephone number is (571) 272-1232 and email address is Yaima.Campos@uspto.gov. The examiner can normally be reached on Monday to Friday 8:30 AM to 5:00 PM.

IMPORTANT NOTE

30. If attempts to reach the above noted Examiner by telephone or email are unsuccessful, the Examiner's supervisor, Mr. Donald Sparks, can be reached at the following telephone number: Area Code (571) 272-4201.

The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300. Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

February 9, 2006

Yaima Campos
Examiner
Art Unit 2185

DONALD SPARKS
SUPERVISORY PATENT EXAMINER